A Model for Meeting Content Storage and Retrieval

Saturnino Luz
Department of Computer Science
Trinity College, University of Dublin
Dublin 2, Ireland
luzs@cs.tcd.ie

Masood Masoodian
Department of Computer Science
The University of Waikato
Hamilton, New Zealand
m.masoodian@cs.waikato.ac.nz

Abstract

This paper presents a model for storage of remote Internet-based multimedia meetings and information retrieval from textual and time-based content. The model builds on a theory of content mapping *that exploits temporal and contextual relationships between media streams. Two prototypes are presented which illustrate the application of the model to a virtual meeting environment, and to a system for visualisation of meeting records on mobile devices. Implications of the proposed content mapping model with respect to interface design and non-linear browsing of time-based media are also discussed.*

1. Introduction

In recent years, the popularisation of low-cost multimedia desktop computers allied with the remarkable growth of the Internet have made networked real-time meetings a common form of communication among geographically remote users. As the cost of physical storage of data decreases, increasingly larger amounts of text, audio and video are being recorded from virtual meetings. Availability of large volumes of data often poses interesting problems. In fact, providing facilities for participants and reviewers of computer-supported meetings to browse, access and structure recorded meeting contents has been an active area of research in computer supported cooperative work (CSCW) for well over a decade. Despite advances in interface and indexing technologies, designing effective recording and information retrieval tools for multimedia meetings remains a particularly challenging task. As [6] point out, although it will soon be possible to record all the meetings a person is ever likely to attend in his or her entire life-span, the usefulness of such data would be limited due to the lack of adequate indexing and retrieval mechanisms.

Understanding how humans might use meeting records is key to building successful interfaces for multimedia access. The time-based nature of such records would suggest a *tape recorder* as the most natural interface metaphor. However, recent studies have shown that given a task of retrieving information from recorded (audio) meetings users prefer to "salvage" selected parts of the recording rather than play it back sequentially [11]. This behaviour somewhat resembles the way people use pen-and-paper minutes, agendas, task allocation tables, and other such low-tech tools as memory-augmenting artefacts in traditional meetings.

Although low-tech artefacts can be very effective in face-to-face meetings, virtual meeting scenarios place extra strains which tend to impair one's ability to keep effective records and take part in a meeting at the same time. This is sometimes referred to as *the divided attention problem* [18]. On the other hand, the technology underlying Internet-based virtual meetings allows for those meetings to be unobtrusively recorded in great detail. The challenge therefore lies in structuring the recorded data in a way that supports natural post-meeting information "salvaging" (retrieval).

Most approaches to meeting storage and indexing to date have focused on modality translation, specially speech to text, as the main means of structuring meeting content [17, 12]. This paper presents a novel approach which encompasses temporal and contextual elements, and pays special attention to relationships among the different media streams in a virtual meeting record. These relationships are organised in a general model for meeting storage and retrieval called *content mapping* [10]. In what follows we discuss related work, describe our content mapping model, outline its storage requirements, describe applications for meeting recording and information retrieval based on this model, and present initial evaluation results.

1.1. Related work

Collaborative systems have been proposed, including copresence systems [4], which enhance traditional memory functionality by supporting the production of *notes*. Notes can be personal or shared *text artefacts* which are offered for discussion in real-time as part of the collaborative pro-

Proceedings of the 11th International Multimedia Modelling Conference (MMM'05)
1550-5502/05 $20.00 © 2005 **IEEE**

cess. At the end of a meeting, each participant typically walks away with a textual record produced by synchronising, merging, pruning and improving such notes. This process is often mediated by speech and face-to-face communication. Textual records usually take the form of minutes and action tables. Such records can be called *static records*, due to the fact that they place greater emphasis on meeting "outcomes" (through persistent and parallel modalities) rather than the processes by which such outcomes were attained. Process records are usually lost along with the transient, sequential speech modality that mediates the process.

A number of meeting browser projects have been proposed which attempt to remedy this situation [13], including the one described in this paper. Any attempt at salvaging transient and sequential records will need to overcome the main obstacles posed by the nature of the modalities involved. While recording the whole audio track of a series of meetings for sequential presentation at a later time proves ineffectual [11] for large volumes of multimedia data, neglecting speech exchanges in favour of static records is clearly unsatisfactory.

An ambitious approach has been pursued which places great emphasis on speech recognition and natural language engineering techniques [17, 12]. This approach involves developing speech recognisers capable of coping with noisy environments and spontaneous speech, detecting communicative acts, and tracking prosody, gestures and facial expressions. State-of-the art speech recognisers have word error rates which vary from around 20 to 60% for Large Vocabulary Conversational Speech Recognition (LVCSR) tasks, depending on recording conditions [13]. Other tasks involved in this kind of approach can be just as demanding as LVCSR, obtaining similarly low accuracy levels. In [16], for instance, a system is described which attempts to recognise *dialogue acts* (i.e. to group utterances into classes such as *statements*, *yes-no-question*, *wh-question*, *quotation*, etc) in spontaneous speech. The maximum accuracy achieved by their system was 65% for automatically recognised words (compared to a chance baseline accuracy of 35%). The issue, however, is not circumscribed to recognition accuracy. Human factors also play a crucial role. Even if recognition and dialogue labelling were perfect, there is no guarantee that users of a meeting browser based on labelled dialogue acts would be able to use those labels effectively for information retrieval. Available evidence actually suggests the opposite. When dialogue act classification is done by humans, inter-annotator agreement is far from perfect ([16] report an 84% rate for tagging by linguistics students.)

Recent work [6, 7] has focused on contextual annotations rather than raw textual data (original or transcribed) as the main vehicle of information retrieval. These approaches build on the enhanced capabilities and widespread availability of mobile devices.

The model presented in this paper aims at bridging the gap between content and context-based approaches by exploiting aspects of time-based media which have received little attention so far. Time-based media have been predominantly manipulated through interfaces designed for linear access to sequential data. The paradigmatic time-based interface exploits the sequential nature of its data by employing a "tape recorder metaphor" and building upon it a set of media specific improvements, such as speech skimming [1] and other modes of search. Although these improvements help overcome some of the limitations imposed by sequential access, the browsing activity itself remains structured along a single dimension, that is, it remains essentially linear. As we have seen, in many cases, such as recordings of meetings, but also of lectures, and broadcasts, time-based media subsume static data which become bound by the same temporal constraints as the former. If the components of such multimedia recordings can be treated as separate *streams* [3], text and graphics can add valuable structural information to time-based media. This is the main aspect of multimedia meeting storage and retrieval explored by the model proposed in this paper.

2. The content mapping model

Content mapping is a data model which views meeting records in terms of discrete (though time-based) data units called *segments*. It departs from usual time-based media in that it facilitates non-linear access to information by grouping segments into *temporal and contextual neighbourhoods*. The concepts of temporal and contextual neighbourhoods help bridge the gap between essentially orthogonal modalities such as speech and text, which are present in most virtual meetings. Although the basic framework of content mapping can be generalised to any number of synchronous data streams, including video, and graphics, this paper focuses on its use in a particular type of multimedia recording that consists of speech and text streams.

We first describe the typical scenarios in which such records are produced. A detailed description of content mapping is presented in section 2.1. For the purposes of this paper, we define a *speech-and-text meeting* as a computer-mediated, synchronous writing task supported by a speech channel. Records of speech-and-text meetings constitute prototypical data handled by the model presented below. The distinguishing features of speech-enabled, synchronous collaborative writing are the immediacy of textual interactions, and the ubiquity of transient contributions (e.g. comments, discussions, back channels, etc, usually conveyed by speech). The general case of speech-and-text meetings therefore includes process models such as the ones described in [14], where combinations of speech and text play a central role in the interaction process.

2.1. Content neighbourhoods

Given a speech-and-text meeting scenario, two straightforward relations can be defined which entail the concepts of *temporal neighbourhoods* (TN) and *contextual neighbourhoods* (CN). The main assumption underlying these concepts is that recorded text and speech can be clustered into natural segments. Different levels of analysis can determine different types of text and audio segments. This framework can accommodate segments types ranging from those derived exclusively from formatting and markup cues to higher-level groupings yielded by more sophisticated theories of human communication and cognition [17, 16]. Examples of text segments include easily identifiable units such as paragraphs, document sections and items in a list, but also units that are harder to identify, such as summaries and rhetorical structures. Similarly, examples of speech segments include (in increasing order of difficulty) audio intervals delimited by silences, communicative turns, and speech acts. As we are mainly interested in investigating relationships between audio and text neighbourhoods rather than specific natural language processing techniques, we leave the segmentation method unspecified and concentrate on defining TN and CN. These can be described as follows:

Temporal Neighbourhood a segment of audio recording is in temporal neighbourhood of a text segment if that audio segment (i) was recorded while the section was being created, changed, or discussed by two or more participants, or (ii) is in a temporal neighbourhood of a related text segment. There could be multiple audio segments in a temporal neighbourhood of a segment, each corresponding to different time intervals during which that section was *active*.

Contextual Neighbourhood a segment of audio recording is in contextual neighbourhood of a document segment when it shares a number of *features* (e.g. keywords) with that segment. As with TNs, a text segment can have multiple audio segments in its contextual neighbourhood.

These notions can be formalised as follows. Given a set $T = \{t_1, ..., t_{|T|}\}$ of text segments, and a set of speech segments $S = \{s_1, ..., s_{|S|}\}$, temporal neighbourhoods are determined by interval overlap as shown in Definition 1.

Definition 1 *A* temporal text-audio mapping *is a function* $tn : T \rightarrow 2^S$ *defined as follows*

$$tn(t_i) = \{s_j : \; t_i^s \leq s_j^s \leq t_i^e \vee t_i^s \leq s_j^e \leq t_i^e \\ \vee s_j^s \leq t_i^s \leq s_j^e \vee s_j^s \leq t_i^e \leq s_j^e\}$$

where t_i^s *and* t_i^e *denote the start and end time of segment* t_i*.*

If we let $keyword(t_i, t_k)$ indicate that segments t_i and t_k share at least one keyword (or phrase), $tn(t_i)$ could be extended to recursively include intervals s_j such that $keyword(t_i, t_k) \wedge s_j \in tn(t_k)$, thus implementing a form of text clustering. Once tn has been constructed, one can also retrieve specific text segments using audio as a starting point by simply inverting the mapping, or defining an *audio-text mapping* $tn_a : S \rightarrow 2^T$, such that:

$$tn_a(s_i) = \{t_j : s_i \in tn(t_j)\} \quad (1)$$

Similarly, one can describe a contextual text-audio mapping cn as in Definition 2. The definition of its audio-text counterpart cn_a is analogous to that of equation (1).

Definition 2 *A* contextual text-audio mapping *is a function* $cn : T \rightarrow 2^S$ *defined as follows:*

$$cn(t_i) = \{s_j : keyword(t_i, s_j)\}$$

where $keyword(t_i, s_j)$ *denote pairs of text and audio segments which share at least one keyword/phrase.*

The relation $\mathcal{T} \subseteq S \times T$ induced by tn is what we call a temporal neighbourhood. A contextual neighbourhood is a relation $\mathcal{C} \subseteq S \times T$ induced by cn.

3. Supporting speech-and-text meetings

In order to explore the possibilities of the model above we have built a basic infrastructure for recording and processing speech-and-text meetings. The basic requirements of such infrastructure are:

- it must have detailed logging capabilities with respect to collaborative text editing,
- it must provide reliable, real-time, multi-party audio communication,
- its audio storage component must have built-in support for speaker (source) identification, and
- it must allow efficient retrieval of relations between timestamped text operations and speech segments.

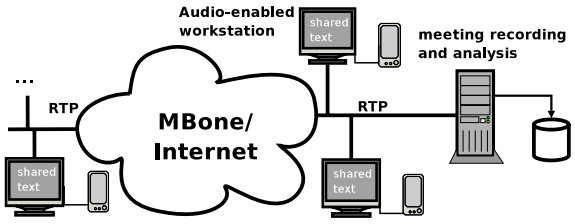

Figure 1. Meeting recording architecture

The audio channel requirements have been met through the use of multicast technology and the *real time protocol* (RTP) [15]. RTP defines mechanisms for encoding source (speaker) identification and timestamping at the client side as part of its packet delivery functionality. Our system uses that information to structure the speech recording. The system architecture (see Figure 1) includes an RTP recorder

which acts as a passive multicast client and writes incoming RTP packets directly to a hash table as they arrive. After the meeting, these packets can be re-sent to a network port upon request from a remote audio client (thus replaying the audio exactly as it was produced in the meeting), or transcoded locally for further processing.

The basic architecture also encompasses a text interaction medium. A real-time collaborative text editor has been implemented which keeps a detailed record of the shared text, and the actions performed on each text segment by each collaborator along with timestamps which indicate exactly when each action was performed. The editor employs an optimistic locking strategy and regards each line-delimited segment as an abstract data unit. Recordable text events include *pointing* gestures, as well as *editing actions* such as *insertions*, *deletions* and *modifications*. Pointing gestures are represented in the interface by a *telepointer* widget. As gestures can (and usually do) span several text segments, they are an invaluable source of information as regards temporal and contextual neighbourhoods. Gestures correspond to about 55% of all text events recorded in our user trials so far. This supports our hypothesis that, in speech-and-text meetings, text acts as a focal point for meeting activity, and therefore can be regarded as a natural starting point for multimedia information retrieval. Simple enhancements in the way text interaction is recorded such as segmentation and activity timestamping can greatly improve access to other (primarily time-based) media.

The recording server stores text in XML format. A fragment of text produced during a collaborative meeting is shown in Figure 2. Both text and interaction history are stored in a single file. Applications typically parse this basic representation format and convert it into more efficient data structures in order to perform content mapping.

```
<?xml version="1.0"?>
<!DOCTYPE comapdoc SYSTEM "file:comapdoc.dtd">
<?xml-stylesheet type="text/css" href="comapdoc.css"?>
<comapdoc>
<meeting date="20030304">
  <description>
Student-supervisor meeting: User testing of a visualisation tool ...
  </description>  <!--participant details omitted-->
</meeting>
<section level="1">
  <segment id="1">
    <header level="1">
    <timestamp agent="A" action="insert" start="35" end="36"/>
Aim of Testing
    </header>
  </segment>
  <segment id="2">
    <timestamp agent="A" action="insert" start="35" end="36"/>
    <timestamp agent="A" action="point" start="83" end="86"/>
    <timestamp agent="A" action="insert" start="167" end="183"/>
    <timestamp agent="A" action="point" start="208" end="210"/>
    <timestamp agent="A" action="point" start="473" end="477"/>
Is the mobile visualization an improvement over a simple text based itinerary?
      (simple conventional paper-base)
            (clarify what is being compared!)
  </segment>
  <segment id="3">
    <timestamp agent="A" action="insert" start="35" end="36"/>
    <timestamp agent="A" action="point" start="235" end="238"/>
    <timestamp agent="B" action="insert" start="247" end="257"/>
    <timestamp agent="A" action="point" start="437" end="446"/>
    <timestamp agent="A" action="point" start="461" end="465"/>
Hypothesis: Visualization once understood by user allows the user to do all tasks done
by text method just as well but also allows user to make estimates and determine how
events interrelate to each other in ways that a text only interface could not.
  </segment>
```

Figure 2. Annotated shared text excerpt

4. COMAP

Two prototypes have been implemented which illustrate the model and overall architecture described above. One of these prototypes is the *COntent MAPper* (COMAP) for desktop computers. COMAP fully supports audio and text recording functionality as described in section 3 and the content mapping framework presented in section 2.1.

Implementing TN mapping is relatively simple. Unlike most existing approaches to meeting browsing which depend on reliable speech recognition [17, 12] or manual tagging [11], a TN-based approach relies solely on the the storage model described in section 3.

CN inference, on the other hand, demands a more complex approach. COMAP, however, does not attempt to handle full meeting transcription (automatic or otherwise). Practical problems apart, we have opted for focusing our research on the structure of the relationship between communication modalities rather than on natural language or speech processing techniques. We regard those techniques as being somewhat complementary to the current system and plan to integrate them in future versions. Currently, COMAP uses a simpler combination of keyword extraction and automatic word spotting as a basis for CN mapping.

Obviously, not all words (and phrases) can be regarded as keywords. If they could, nearly all text segments would be interrelated, rendering CN mapping useless. In information retrieval terms: all relevant segments would be *recalled* but the user would be overloaded with information of very low *precision*. In order to select the most relevant words and phrases we use a module comprising part-of-speech tagging (POS), stop-word removal, collocation analysis and feature extraction. POS tagging assigns each word in the text a grammatical category. This phase precedes stop-word removal and serves as a pre-processing stop to collocation analysis. We employ the transformation-based algorithm of [2] which yields an overall tagging accuracy of around 96.5% for POS tagging.

Stop-word removal consists simply of table lookup and eliminates very common and closed class words, such as determiners, auxiliaries, conjunctions, etc. Collocation analysis aims at finding phrases which may be selected as representative features of a text segment. The approach used in this module is a POS filtering algorithm adapted from [5]. This algorithm consists of selecting POS sequences that are likely to form phrases. For English, good candidate patterns include nouns followed by nouns (e.g. “speech recognition”), adjectives followed by nouns or proper nouns

(e.g. “passive RTP”), adjectives followed by adjectives and nouns (e.g. “Gaussian random variable”), and many others.

Finally, the problem of selecting those terms that best characterise a text segment can be recast as *feature selection*. The task can be described as reverse classification, where each text segment represents a category, and one wishes to find the features which best describe it. The output of the information extraction module is a *word table* containing keywords and key phrases, along with the text segments in which they occur. CN-based meeting indexing is largely a matter of cross referencing word tables and the timestamps extracted from audio and text media records.

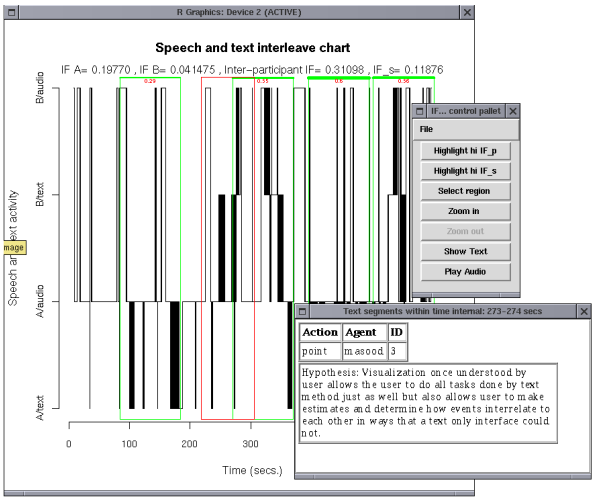

Figure 3. The COMAP browser

Figure 3 shows the main COMAP user interface components. The user is initially presented with a timeline depicting the entire meeting and patterns of media activity (audio and text) per participant. The user can select segments to display or playback based on CNs and TNs visible on this graphical representation of the meeting. Since there are typically many speech segments associated with a text segment (and vice versa), we have devised an *Interleave Factor* (IF) metric through which we rank TNs and CNs in order of relevance. IF is calculated with respect to the likelihood of co-occurrence of speech and text events, seen as a continuous probability distribution. The key observations on which it is based are that concurrent events are often semantically related, and that intervals with the greatest levels of text-audio inter-participant activity are the most likely to contain relevant information (see [8] for details). IF-ranking is particularly useful for meeting browsing on small screens, as implemented in the system described below.

5. A system for mobile devices

Although COMAP is useful as a general platform for exploring content mapping, a desktop-based meeting browser would be of limited usefulness in real-world situations. Current everyday work practices often require professional people to be working not only when they are at their offices, but more increasingly while they are travelling or when they are at home. Access to information is no longer considered a luxury, rather it is often crucial for ones ability in carrying out work responsibilities. It is not, therefore, surprising that mobile computing and communication technology are being adopted and used for accessing information while *on the move*. The range of activities and tasks for which such devices are being used is also growing rapidly. This growth is further fueled by increasing processing power, networking, and input-output capabilities of handheld devices such as mobile phones, PDAs, and ever smaller laptops.

Despite this increasing demand for more useful applications which allow mobile access to personal and work-related data, the range of such applications is at present limited to email and the WWW. Although most PDA type devices are equipped with standard office applications, including word-processors and spreadsheets, these applications are minimal forms of their desktop counterparts which have not been specifically developed for small devices (with their inherent limitations), making them less usable.

Therefore, developers of applications for meeting content storage and retrieval need to keep in mind the requirements of mobile people who might want to access recorded meeting contents from their handheld devices. It is also important to note that to satisfy these requirements, meeting content storage and retrieval systems have to be adaptable to mobile devices with their limitations and capabilities. This adaptability means that the data should be recorded and indexed for use in a device with small data storage, and slower processing power than conventional desktop computers often used by existing information retrieval systems.

5.1. HANMER

To demonstrate the suitability of the content mapping model for devices with restricted processing power and output displays, we have developed a prototype system called HANMER (HANdheld Meeting browsER). This prototype, which has been more fully described in [9], complements COMAP by allowing users to download recorded meeting contents onto PDAs so that they can retrieve meeting-related information they require while on the move.

From a retrieval perspective, HANMER provides the same functionality as COMAP. Both systems utilise content mapping between synchronised audio and textual streams recorded during online meetings. These meetings are al-

ways conducted on desktop computers where participants have access to full-duplex audio and a shared text editor, as described above. After the meeting, however, the recorded data undergo further processing prior to downloading by HANMER. The data format used by HANMER and COMAP are also exactly the same, which allows seamless sharing of data between desktop and PDA users.

Despite the fact that both COMAP and HAMNER provide the same functionality and utilise the same underlying indexed data, the design of the COMAP system and its interface had to be modified to make it more suitable for a PDA device with limited input and output capabilities. We have developed two alternative meeting content visualisation techniques for HANMER, one of which is similar to that of COMAP, while the second is considerably different. The alternative design allows compact visualisation of temporal relationships between audio and textual meeting contents on the small display area of a typical PDA. Both of these visualisations are briefly described here.

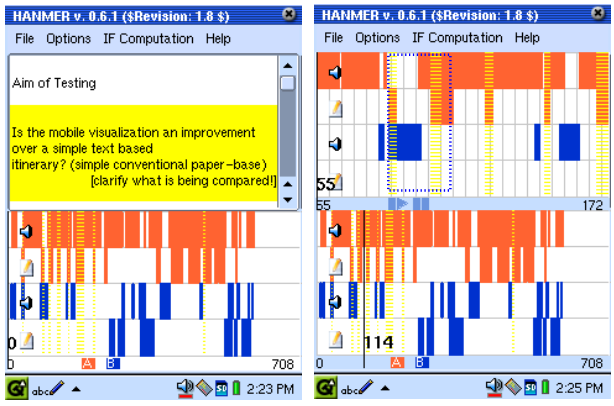

Figure 4. Basic visualisation on HANMER

Figure 4 shows the first visualisation style. The user would generally start browsing meeting records by reading a text segment produced during the meeting (shown in the top area of Figure 4, (a)). This reading of text may be followed by playing back of audio segments in the temporal neighbourhood of that text segment. In this particular visualisation, temporal neighbourhoods are shown using a time series of audio and text events plotted as a step function whose values range from 0 to the number of participants, times the number of media streams. Initially, the user is presented with a radar view of the entire meeting, as shown in the bottom area of Figure 4. The user can then navigate to a specific part of the meeting by selecting the required section from the radar view, which can then be examined in more detail (top area of Figure 4, (b)), followed by retrieval of text segments, or audio playback. Generally speaking, navigation will be guided by a text component, and if the user is interested in a subject referring to a particular text segment, the segment may then be selected, which in turn causes all intervals in its temporal neighbourhood to be highlighted on the meeting profile. This narrows the search down to a few segments which may also be in other temporal neighbourhoods whose audio and text segments can be accessed through their visual representation.

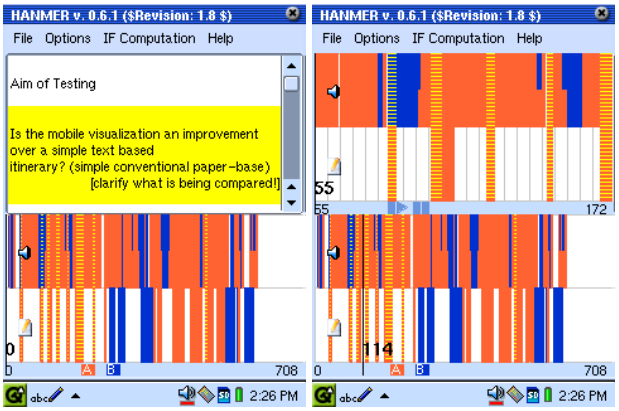

Figure 5. Mosaic-style visualisation

This visualisation style is limited due to the fact that it assigns a horizontal timeline to each of the media streams (audio, text) and each of the participants, in both the overview of the meeting (bottom part of the visualisation) as well as the detailed view of the selected meeting segment (top part of the visualisation). The obvious restriction with this style of visualisation is that as the number of participants and/or recorded media streams increases it becomes rather difficult to display all the timelines on the screen.

Figure 5 shows a visualisation designed to overcome this limitation. It assigns a single timeline to each of the media, and then combines and presents the contributions of all participants for that particular media across this single timeline. Using this method, the number of timelines that need to be displayed is limited to the number of media streams recorded. The space allocated to a particular timeline is used proportionally to show the contributions of each of the participants in terms of that medium. For instance, if only one participant, say, person A, is talking for a period of time then the audio timeline corresponding to that time period is covered by a rectangle which has the colour allocated to that participant (e.g. orange). However, if both participants A and B are talking simultaneously for a period of time, then that time period is divided vertically into two, and covered with rectangles that have the colours associated with A and B (e.g. orange and blue). This allows the number of participants that can be depicted on the visualisation to be increased to a handful, which corresponds to the maximum group size our system aims to support.

One of the benefits of this new visualisation is that it allows users to easily locate parts of the meeting which have a higher degree of concurrent activities (i.e. high IF areas), as these correspond to the parts which are more colourful and look like mosaics. Furthermore, a quick look at the visualisation would also easily show if any of the participants are more active than the others during the meeting, simply because their colour would be more dominant in the graph.

Although the second visualisation is more suitable for devices with limited display area, particularly when the number of meeting participants and/or media type is larger than two, the first visualisation is comparable for meetings in which the number of meeting participants and media types is small. Usability testing is required to identify the effectiveness of each of these visualisations. Currently HANMER allows the users to dynamically choose which visualisation is used for accessing the recorded meeting data.

6. Conclusions

This paper presented a model for multimedia storage and retrieval of meeting data which builds on the concept of contextual mapping.

Although the prototypes we have developed to illustrate this model deal only with text and speech, the proposed framework can accommodate other time-based media such as video. We are currently investigating the possibility of incorporating recorded video from collaborative online meetings into our indexing and retrieval system. These video recordings could come from a number of different sources, such as normal video conferencing tools, or screen capture technology recording the computer display of the meeting participants. The idea is that recordings of the computer screens and user interface events captured during the meetings could be synchronised with audio recordings to allow browsing of online meeting contents which do not necessarily focus around textual documents, as is the case with our existing system. Predominance of non-textual interaction is particularly common in online training meetings, where for example demonstration of computer software is the main objective of the meeting.

Clustering and visualisation techniques which build on the content mapping model but depart more radically from the timeline metaphor in order to explore the non-linear nature of meeting data are also being investigated.

Acknowledgments

This work has been supported by Enterprise Ireland through a Basic Research Grant.

References

[1] B. Arons. SpeechSkimmer: Interactively skimming recorded speech. In *Proceedings of UIST'93*, pages 187–196, 1993.

[2] E. Brill. Some advances in transformation-based part of speech tagging. In *Proceedings of the 12^{th} National Conference on Artificial Intelligence. Vol 1*, pages 722–727, 1994.

[3] S. Gibbs, C. Breiteneder, and D. Tsichritzis. Data modeling of time-based media. *ACM SIGMOD Record*, 23(2):91–102, 1994.

[4] S. Greenberg, M. Boyle, and J. LaBerge. PDAs and shared public displays: making personal information public, and public information personal. *Personal Technol.*, 3(1), 1999.

[5] J. S. Justeson and S. M. Katz. Technical terminology: some linguistic properties and an algorithm for identification in text. *Natural Language Engineering*, 1(1):9–27, 1995.

[6] N. Kern, B. Schiele, H. Junker, P. Lukowicz, and G. Tröster. Wearable sensing to annotate meeting recordings. *Personal Ubiquitous Computing*, 7(5):263–274, 2003.

[7] D.-S. Lee, B. Erol, J. Graham, J. J. Hull, and N. Murata. Portable meeting recorder. In *Procs. of the 10^{th} International Conference on Multimedia*, pages 493–502, 2002.

[8] S. Luz. Interleave factor and multimedia information visualisation. In *Procs. of Human Computer Interaction 2002*, vol 2, pages 142–146, 2002.

[9] S. Luz and M. Masoodian. A mobile system for non-linear access to time-based data. In *Proceedings of Advanced Visual Interfaces AVI'04*, pages 454–457. ACM Press, 2004.

[10] M. Masoodian and S. Luz. COMAP: A content mapper for audio-mediated collaborative writing. In *Procs. of HCI International 2001*, pages 208–212, 2001.

[11] T. P. Moran, L. Palen, S. Harrison, P. Chiu, D. Kimber, S. Minneman, W. van Melle, and P. Zellweger. "I'll get that off the audio": A case study of salvaging multimedia meeting records. In *Procs. of CHI 97*, pages 202–209, 1997.

[12] N. Morgan, D. Baron, J. Edwards, D. Ellis, D. Gelbart, A. Janin, T. Pfau, E. Shriberg, and A. Stolcke. The meeting project at ICSI. In *Procs. of Human Language Technologies Conference*, 2001.

[13] *NIST Automatic Meeting Transcription, Data Collection and Annotation Workshop*, 2001.

[14] I. R. Posner and R. M. Baecker. How people write together. In *Readings in Computer Supported Collaborative Work*, pages 239–250. 1993.

[15] H. Schulzrine, S. Casner, R. Frederick, and V. Jacobson. RTP: A transport protocol for real-time applications. IETF Internet Draft draft-ietf-avt-rtp-new-04, February 1999.

[16] A. Stolcke, K. Ries, C. N, E. Shriberg, R. Bates, D. Jurafsky, P. Taylor, R. Martin, C. V. Ess-Dykema, and M. Meteer. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational Linguistics*, 26(3):339–373, 2000.

[17] A. Waibel, M. Brett, F. Metze, K. Ries, T. Schaaf, T. Schultz, H. Soltau, H. Yu, and K. Zechner. Advances in automatic meeting record creation and access. In *Procs. of the Intl. Conf. on Acoustics, Speech and Signal Processing*, 2001.

[18] M. Wiberg. Knowledge management in mobile CSCW: evaluation results of a mobile physical/virtual meeting support system. In *Proceedings of HICSS-34*. IEEE Press, 2001.